Attorney Docket No.: 07781.0036
Customer No. 22,852

UNITED STATES PATENT APPLICATION

FOR

METHOD AND APPARATUS FOR IMPLEMENTING A SPEECH INTERFACE FOR A GUI

BY

FRANKIE JAMES
AND
JEFF ROELANDS

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

METHOD AND APPARATUS FOR IMPLEMENTING A SPEECH INTERFACE FOR A GUI

FIELD OF THE INVENTION

[001] This invention relates to the field of graphical user interfaces (GUIs) for PC systems, and, more specifically, to a speech interface for a GUI and a method for interfacing with a GUI using the same.

BACKGROUND OF THE INVENTION

[002] Many business applications use GUIs to allow users to interface with the applications for performing a number of operations. Typically, GUIs are mouse- and keyboard-intensive, which can be problematic or even impossible to use for many people, including those with physical disabilities. One type of interface that avoids a mouse or keyboard is a speech interface. A speech interface allows audio input of commands to communicate with applications, and can be used by anyone who wishes to speak to their system, such as mobile users with inadequately-sized keyboards and pointing devices.

[003] One of the main challenges for a speech interface is specifying the desired target of an audio input, especially in a GUI where multiple selectable objects such as windows, text fields and icons, can have the same label or name. In these situations, it is important for both the computer system and the user to know the current focus when an audio input is issued, to help the system resolve possible ambiguities and to help the user keep track of what he is doing.

[004] One type of ambiguity is called "target ambiguity," where the target of a user's action is ambiguous and must be resolved. In a physical interaction involving a mouse or other pointing device, users specify the target of their actions

LAW OFFICES
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

directly by clicking on the selectable object of interest. Target ambiguities caused by selectable objects that have the same name are resolved through the physical interaction. With audio input, users do not have a way to resolve target ambiguities physically; therefore, the target ambiguity must be handled in some other way.

[005] Traditional speech interfaces for GUIs typically emulate the keyboard and mouse directly using spoken equivalents; however, they are slow to operate and often take quite a bit longer to select an object than conventional mouse or keyboard techniques. Conventional speech interfaces lack public acceptance due to inaccurate control of the interfaces.

[006] Other traditional speech interfaces for GUIs combine audio input with alternative pointing devices such as head- or eye-tracking technologies. However, conventional alternative pointing devices require calibration and expensive equipment, making them difficult to set up and use on computers shared by multiple people.

[007] Still other traditional speech interfaces provide object selection solely by audio input. These speech interfaces explore a current window or current screen area to find selectable objects that match the audio input. One limitation of these speech interfaces is that they only explore the current screen area to find selectable objects matching the audio input. They do not explore the other screen areas for a match to the audio input. Instead, additional audio inputs are required to look for matches to the audio input in screen areas other than the current screen area. These additional required audio inputs decrease the efficiency of conventional speech interfaces.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

[008] Another limitation of these traditional speech interfaces involves their capability to resolve target ambiguity. These interfaces mark selectable objects matching the audio input with opaque icons for subsequent selection. However, placing an opaque icon adjacent to the selectable object often pushes screen elements out of place and distorts the screen layout. Furthermore, overlaying the selectable object with an opaque icon often obscures the underlying text and graphics. Finally, displaying opaque icons for all objects that match each other often clutters the screen. Thus, traditional speech interfaces often fail to maintain the integrity of screen layout and the view of the text and graphics of the selectable objects when resolving target ambiguities.

SUMMARY OF THE INVENTION

[009] Methods and apparatus consistent with the present invention provide a speech interface for a GUI via a computer system.

[010] Consistent with the present invention, a method for providing speech control to a GUI containing objects to be selected includes dividing the GUI into a plurality of screen areas; assigning priorities to the screen areas; and receiving a first audio input relating to the selection of one of the objects in the GUI. The method further includes determining one of the screen areas that has the highest priority and includes a first object matching the first audio input; selecting the first object, if the determined screen area only contains one object matching the first audio input; and using a second input to select one of the objects that matches the first audio input in the determined screen area, if the determined screen area contains more than one object that matches the first audio input.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

BRIEF DESCRIPTION OF THE DRAWINGS

[011] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate an embodiment of the invention and together with the description, serve to explain the principles of the invention.

[012] FIG. 1 is a block diagram of a computer system on which methods and apparatus consistent with the invention may be implemented;

[013] FIG. 2 is a block diagram showing components of a computer device which can be used to implement the FIG. 1 computer system;

[014] FIG. 3 is a diagrammatic illustration of a GUI window which can be utilized to implement the method and system of the present invention;

[015] FIG. 4A is a diagrammatic illustration showing a document of HTML data which can be utilized to implement the method and system of the present invention;

[016] FIG. 4B is a diagrammatic illustration of a data structure which can be utilized to implement the method and system of the present invention; and

[017] FIG 5. is a flow diagram of the sequence of operations for providing a speech interface for a GUI in accordance with an example implementation of the present invention.

DETAILED DESCRIPTION

[018] Reference will now be made in detail to exemplary embodiments of the invention illustrated in the accompanying drawings. Wherever possible, the same reference numbers in different drawings refer to the same or like parts.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

A. Overview

[019] Methods and apparatus consistent with the invention provide a speech interface for a GUI. One embodiment includes dividing a GUI for a system into screen areas containing one or more selectable objects, and assigning priorities to the screen areas based on some criteria, for example, usage. The screen area with the highest priority is indicated, for example, by visual output such as highlighting. Next, an audio input is received, and its receipt confirmed, for example, by an audio output. In response to the received audio input, a system determines the screen area having (1) the highest priority and (2) a selectable object matching the audio input. If that screen area only contains one such object, then the system selects that object. However, if that screen area contains more than one matching object, then the system uses a second input to select one of the objects matching the audio input from that screen area.

[020] Using a second input to select one of the objects that match an audio input includes marking the objects that matched the audio input, such as with icons; receiving a second audio input relating to the selection of one of the marked objects; and selecting the marked object that best matches the second audio input. The icons may be semi-transparent and may be laid over the objects. In addition, the icons may include numbered labels and may be removed from the screen area upon object selection.

B. Architecture

[021] FIG. 1 is a block diagram of a computer system 100 on which methods and apparatus consistent with the invention may be implemented. FIG. 1

LAW OFFICES
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

shows, for example, a client/server computer system wherein a client computer 130 is connected to a Web server 110 via a network 120. Network 120 may be a local area network (LAN), a wide area network (WAN), or a large multi-user network like the Internet. In one embodiment, the Web server 110 is also connected to a backend server 105. Computer system 100 is suitable for use with the C++ programming language, although one skilled in the art will recognize that methods and apparatus consistent with the invention may be applied to other suitable user environments.

[022] FIG. 2 shows a block diagram of components of a computer which could be used to implement the client or server computers in Fig. 1 computer system 100. Computer 200 includes several components that are all interconnected via a system bus 230. Computer 200 communicates with other user's computers on network 120 via a network interface 220, examples of which include Ethernet or dial-up telephone connections.

[023] Computer 200 contains a processor 205 connected to a memory 225. Processor 205 may be a microprocessor or any other micro-, mini-, or mainframe computer. Memory 225 may comprise a RAM, a ROM, a video memory, or mass storage. The mass storage may include both fixed and removable media (e.g., magnetic, optical, or magnetic optical storage systems or other available mass storage technology). Memory 225 may contain a program, such as a web browser, and an application programming interface.

[024] A user typically provides information to computer 200 via input devices 215, such as a conventional microphone. In return, information is conveyed

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

to the user via output devices 210, such as a conventional visual display device and an audio display device. Output devices 210 and input devices 215 connect to computer 200 via appropriate I/O ports. The input and output devices may be physically connected to system bus 230, or could communicate via some other connection, such as by use of an infrared or other wireless signal.

[025] The client computer 130 may run a "browser" 135, which is a software tool used to access a Web server 110. A Web server 110 operates a web site which supports files in the form of documents and pages. Browser 135 sends out requests to Web server 110 via network 120, and Web server 110 returns responses to browser 135 via network 120. Browser 135 acts upon those responses such as by displaying content to the user. The content portion of the responses can be a "Web page" expressed in hypertext markup language (HTML) or other data generated by the Web server 110.

[026] One exemplary computer system 100 that can implement the present invention is the SAP R/3™ system. In that system, a backend server 105, such as an SAP R/3™ server, contains program applications that can be performed on client computer 130. Web server 110, such as an Internet Transaction Server (ITS) from SAP, reads the application content from the backend server 105 and translates the application content into HTML.

[027] An extension 115 in the Web server 110 structures the HTML content. The extension 115 is an application program run on Web server 110 that structures HTML content by, for example, defining control groupings and data types within the HTML content. Extension 115 adds data to the HTML content via the form

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

of traditional meta-data tags or XML type structured data. The extension 115 may be written in programming languages such as Javascript™, Visual Basic™, C++ and C.

[028] On the client side, in the exemplary system, the client computer runs Windows™ O/S to create a multi-framed environment. Browser 135 receives the structured HTML content from the Web server 110 via network 120. An extension module 140 within browser 135 then voice-enables the received HTML content. Extension module 140 consists of program applications to be performed on the client's processor 205 for voice-enabling the HTML content. In the exemplary system, the program applications include a parser 145, a prioritization module 150; a speech recognition engine 155; an event handler 160; and a change detector 165. These applications may be written in languages such as Javascript™, Visual Basic™, C++ and C.

[029] Extension module 140 reads in the structured HTML content and identifies elements in the content and their location within the content structure. Parser 145 then parses each element using standard parsing techniques, such as DOM-parsing in Javascript™. For instance, upon identification of a group of two items, such as a label tag and text data, parser 145 extracts text data out of the label. The extracted text data can include identifying text, an object identification, an event and an action associated with an the element (e.g. an action associated with an on-click event). Extension module 140 then assigns a voice command, such as the identifying text, for the element.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

[030] Extension module 140 then loads the element data into a prioritization module 150. Prioritization module 150 stores element data, such as the associated voice command, the object identification, the event, and the action in a data structure consisting of categorized areas. The data structure and categories correspond to the structure and groups of the originally read content. By storing the element data in categories, or grammars, the prioritization module 150 can associate more than one element.

[031] As an illustrative example, FIG. 3 shows an exemplary GUI 300 displaying content generated by a server to the user. GUI 300 consists of five screen areas. On the left, GUI 300 shows the "In Use" area 315, the "Workplace Favorites" area 320, and the "Roles" area 325. On the right, GUI 300 shows two areas, the "Departing" area 305 and the "Arriving" area 310. FIG. 4A depicts an example of HTML content 400 associated with "In Use" area 315 of FIG. 3. After extension module 140 reads through the HTML elements associated with the objects contained within "In Use" area 315, it loads the parsed element data into the "In Use" category, or grammar, in prioritization module 150. FIG. 4B shows an example of grammar structure 450 in prioritization module 150 having categorized element data.

[032] After storing the element data into grammars, prioritization module 150 assigns a priority value to each of the grammars. This effectively assigns a priority value to each of the elements contained within the grammars. For example, in FIG. 4B, assigning the highest priority to the "In Use" grammar effectively assigns the highest priority to each element contained within the "In Use" grammar. Client

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

computer 130 uses this prioritized element data to respond to an audio input relating to a selection of an object contained within the GUI.

[033] Client computer 130 also contains a standard speech recognition engine (SR engine) 155, such as a Microsoft™ SAPI-compliant SR engine. In one embodiment, extension module 140 contains SR engine 155. SR engine 155 produces a text stream from an audio input using dictionaries or phonetic recognition patterns. In one embodiment, client computer 130 loads or registers the stored voice commands against a standard SR engine 155. SR engine 155 then produces a text stream that matches a registered voice command. Extension module 140 uses the text stream to determine the selected object as enumerated below.

[034] The selectable objects may have actions or functions associated with them that are performed upon their selection, such as the opening or closing of an application, or an action within the screen area. In one embodiment, extension module 140 contains an event handler 160 to enable the performance of an operation related to the speech interface prior to performing the action associated with an object selection. Event handler 160 replaces the function, or action, with an extension module function for performing an operation related to the speech interface. The extension module function is then set to call the previously replaced function upon completion of the operation related to the speech interface. For example, upon selection of a button, extension module 140 could produce highlighting prior to performing the action associated with the button selection. In another embodiment, event handler 160 also enables the performance of an operation related to the speech interface after performing the action associated with

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

an object selection. For instance, upon selection of a button, extension module 140 could adjust screen area priorities after performing the action associated with the button selection.

[035] In another embodiment, extension module 140 also contains a change detector 165 to monitor a screen area's content and to detect changes so that prioritization module 150 can account for the elements associated with a GUI. For instance, upon the selection of a button, change detector 165 would detect changes made to the content of a screen area. Upon detecting changes, parser 145 would re-parse the content and prioritization module 150 would update data structures for that screen area.

[036] FIG. 5 depicts a flow diagram 500 of a sample sequence of operations for providing a speech interface for a GUI which are consistent with the present invention. In one embodiment, computer 130 creates a GUI containing selectable objects by using HTML. First, the system divides the GUI into screen areas (Step 505). Next, the system assigns priorities to the screen areas (Step 510). After receiving a first audio input (Step 515), the system determines the highest priority screen area that contains an object matching the first audio input (Step 520). Next, the system determines whether that screen area contains only one object that matches the first audio input (Step 525). If so, the system selects the matching object (Step 545). If not, the system marks each matching object within that screen area with a marker (Step 530). The system then receives a second audio input (Step 535) and determines whether the second audio input matches one of the markers (Step 540). If so, the system selects the object marked by the

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

matching marker (Step 545). If not, the system determines the highest priority screen area containing an object matching the second audio input (Step 520).

[037] As explained above, the system divides the GUI 300 into five screen areas: "Departing" area 305, "Arriving" area 310, "In Use" area 315, "Workplace Favorites" area 320, and "Roles" area 325. Prioritization module 150 assigns the screen areas a priority. For instance, "Departing" area 305 could have the highest priority, "Arriving" area 310 could have second priority, "In Use" area 315 third priority and so on. There are two objects marked "Friday," one within "Departing" area 305 and one within "Arriving" area 310, and three objects marked "Airlines," two within "In Use" area 315 and one within "Roles" area 325.

[038] If, for example, the user says "Friday," the system first looks for a matching object in the highest priority area, "Departing" area 305. Since there is an object matching "Friday" here, the object is selected and the corresponding action is performed for that object. Although an object matching "Friday" also resides within the "Arriving" area 310, the "Departing" area 305 is the screen area from which the object is selected because "Departing" area 305 has a higher priority than "Arriving" area 310.

[039] If, on the other hand, the user says "Airlines," the system first looks for a matching object in the highest priority area, "Departing" area 305. Since no object matches "Airlines" in that area, the system looks for a match in the next highest priority area, "Arriving" area 310. Again, no object in "Arriving" area 310 matches "Airlines," so the system continues to look for a match in the other areas in order of their priority. The system finally finds a matching object in "In Use" area

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

315. Although an object matching "Airlines" lies within "Roles" area 325, the system selects the object from "In Use" area 315 because the "In Use" area 315 has a higher priority than "Roles" area 325.

[040] There are two instances of "Airlines" in "In Use" area 315, however, so the system presents markers, such as semi-transparent, numbered icons, to allow the user to select one of the instances. These types of semi-transparent, numbered icons may be referred to as 'Representational Enumerated Semi-transparent Overlaid Labels for Voice' (RESOLV) icons from SAP. The user selects one of the "Airlines" objects by saying the number on the associated marker.

[041] As described in detail above, methods and apparatus consistent with the invention provide a speech interface for a GUI. The foregoing description of an implementation of the invention has been presented for purposes of illustration and description. Modifications and variations are possible in light of the above teachings or may be acquired from practicing the invention. For example, the foregoing description is based on a client-server architecture, but those skilled in the art will recognize that a peer-to-peer architecture may be used consistent with the invention. Moreover, although the described implementation includes software, the invention may be implemented as a combination of hardware and software or in hardware alone. Additionally, although aspects of the present invention are described as being stored in memory, one skilled in the art will appreciate that these aspects can also be stored on other types of computer-readable media, such as secondary storage devices, like hard disks, floppy disks, or CD-ROM; a carrier wave from the Internet; or other forms of RAM or ROM.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600

[042] The scope of the invention is therefore defined by the following claims and their equivalents.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT
& DUNNER, L.L.P.
STANFORD RESEARCH PARK
700 HANSEN WAY
PALO ALTO, CALIF. 94304
650-849-6600